

# Bioinformatique appliquée

- Cours 3 -

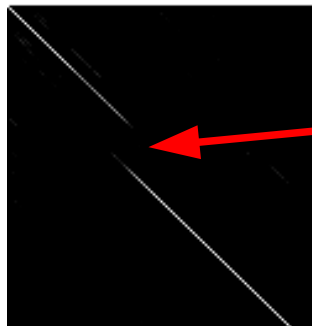
## Alignement de séquences

- principes
- un algorithme d'alignement: programmation dynamique
- alignement global, alignement local

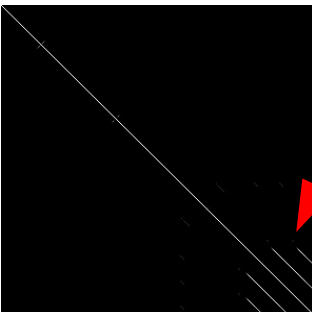
# Les comparaisons de séquences

## • DOTPLOT

- j'ai 2 séquences, est-ce qu'elles ont des choses en commun ?



délétion d'une portion de séquence



répétition d'une portion de séquence

QUALITATIF

## • Alignements de séquences

- j'ai 2 séquences, quel est leur pourcentage de similarité/score ?

Score = 80.9 bits (198), Expect = 6e-15, Method: Compositional matrix adjust.  
Identities = 94/230 (40%), Positives = 120/230 (52%), Gaps = 33/230 (14%)

```

Query  18  DLGLC---KKRPKPGGWNTGGSRYPGQGSPGGNRYPPQGGGGWGQPHGGGGWQPHGGGGW  74
          D+ L   K +P  GGW  G R P          G + P          G PH  G+  PH  G+
Sbjct  20  DVALSKKGKGPSSGGGWAGSHRQPSYPRQPGYPHNPGYPHNPGYPHNPGY--PHNPGY-  76

Query  75  QPHGGGGWQ----PHG---GGWQ-----GGGTHSQWNKPSK-PKTNMKHMAGAAAAGA  120
          PH  G+ Q  PH  GWGQ          GG  H+Q  KP K PKTN KH+AGAAAAGA
Sbjct  77  -PHNPGYPQNPQYPHNPGYPGWGQGYNPSSGGSYHNQ--KPWKPPKTNFKHVAGAAAAGA  133

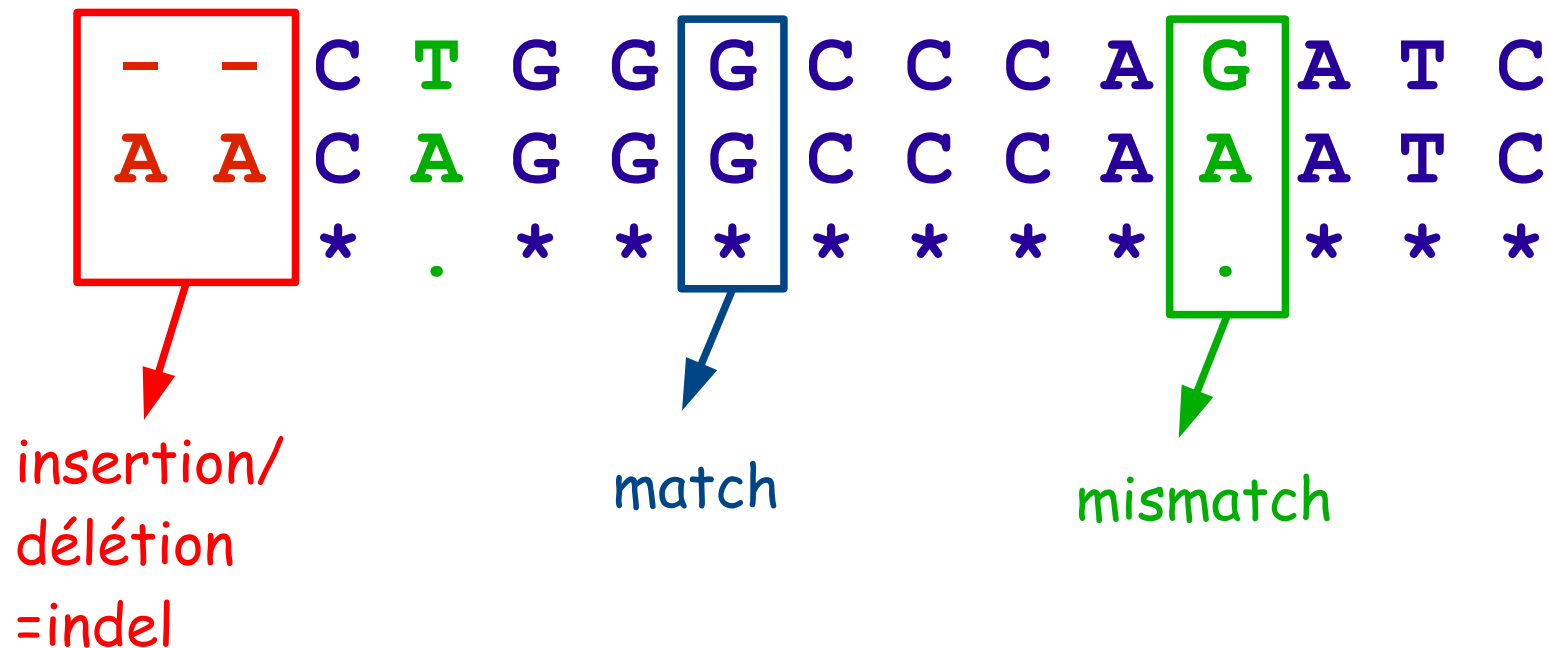
Query  121 VVGGGGLGYMLGSAMSRPIIHFGSDYEDRYRENMRYPNQVYYRPMDEYSNQNNFVHDCV  180
          VVGGGGLGY +G  MS   HF S  E R++ EN  RYPN+VYYR          Q+ FV DC
Sbjct  134 VVGGGGLGYAMGRVMSGMNYHFDSPDEYRWWSENSARYPNRVYYRDYSSVPVQDVFVADCF  193

Query  181 NITIKQHTVTITTK-----GENFTETDV--KMMERVVEQMCITQYER  220
          NIT+ ++++  K          N TE ++  K++ +V+ +MC+ QY
Sbjct  194 NITVTEYSIGPAACKNTSEAVAAANQTEVEMENKVVTKVIREMCVQYRE  243
  
```

QUANTITATIF

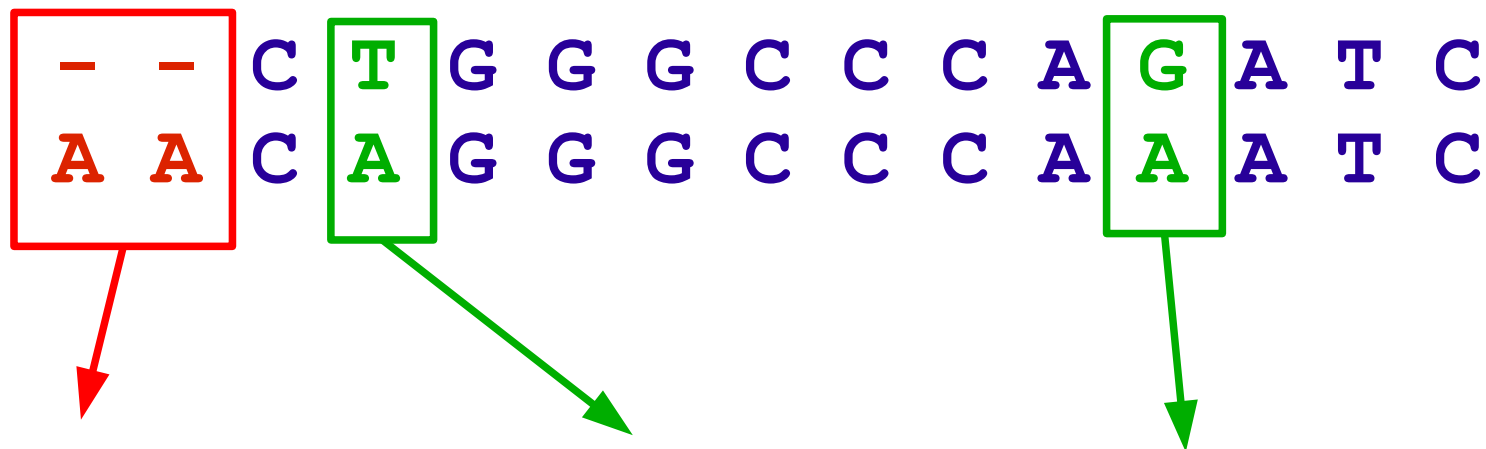
# Alignement de séquences

- soient 2 séquences a priori homologues
  - CTGGGCCCCAGATC
  - AACAGGGCCCAAATC
- voilà un alignement possible



# un alignement raconte une histoire

il était une fois, au cours de l'évolution ...



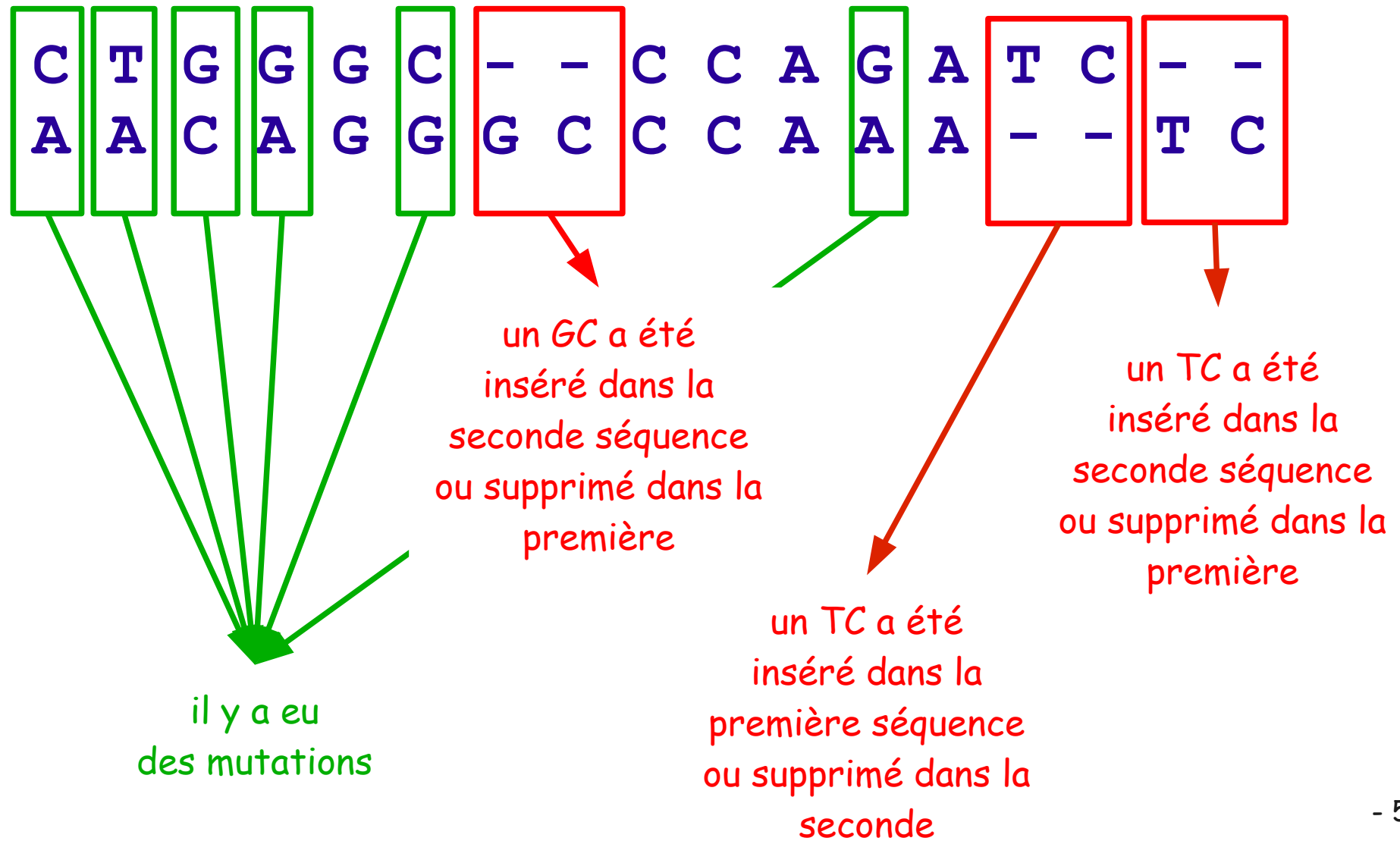
... un AA a été  
inséré dans la  
seconde séquence  
ou supprimé dans la  
première

... un T est devenu  
un A, ou vice-versa

... un G est devenu  
un A, ou vice-versa

# autre un alignement raconte une histoire autre

il était une fois, au cours de l'évolution ...





- - C T G G G C C C A G A T C  
 A A C A G G G C C C A A A T C

Quelle histoire est ce que vous ~~préfèrent~~ ?  
 croyez

C T G G G C - - C C A G A T C - -  
 A A C A G G G C C C A A A - - T C

alignement des  
séquences



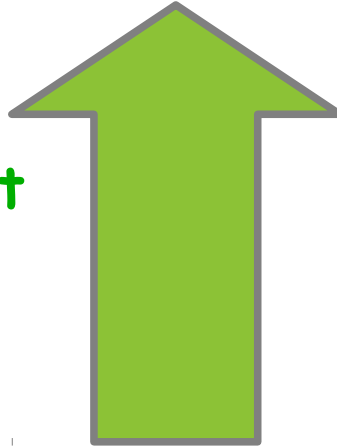
à défaut de connaître la  
vraie histoire, on va  
chercher l'alignement qui  
raconte l'histoire la plus **probable**

histoire évolutive des  
séquences



alignement des  
séquences

et si on connaissait  
l'histoire ?



histoire évolutive des  
séquences



ACTGGGCCCAATC

1 délétion

1 substitution

1 insertion

1 substitution



CTGGGCCCAATC



AACAGGGGCCCAATC

alignement correct

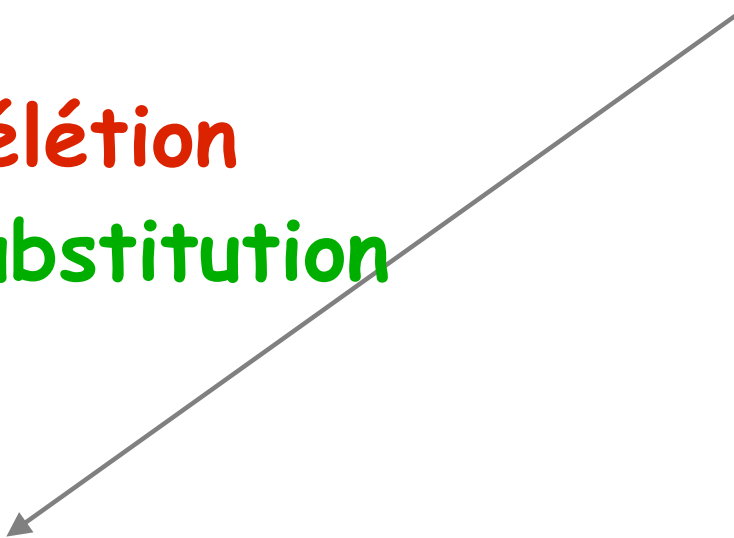
→ celui qui aligne les positions homologues



ACTGGGCCCAATC

1 délétion

1 substitution



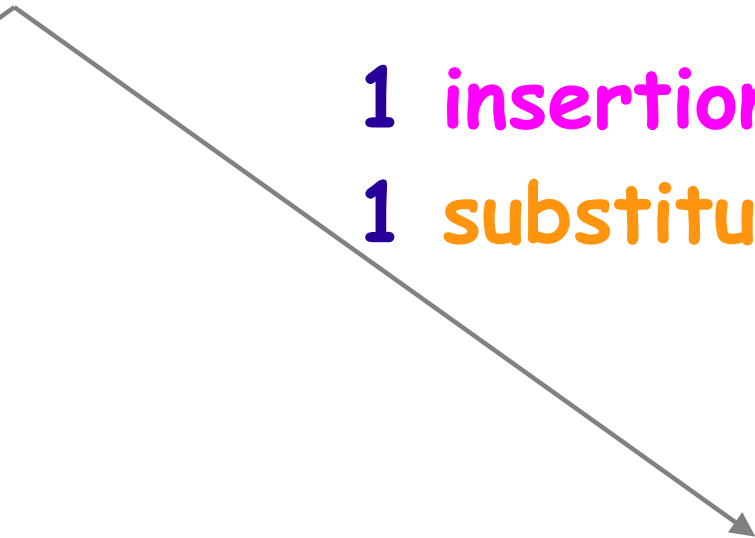
CTGGGCCCAAGATC

Alignement correct

--CTGGGCCCAAGATC  
AACAGGGGCCCAATC

1 insertion

1 substitution

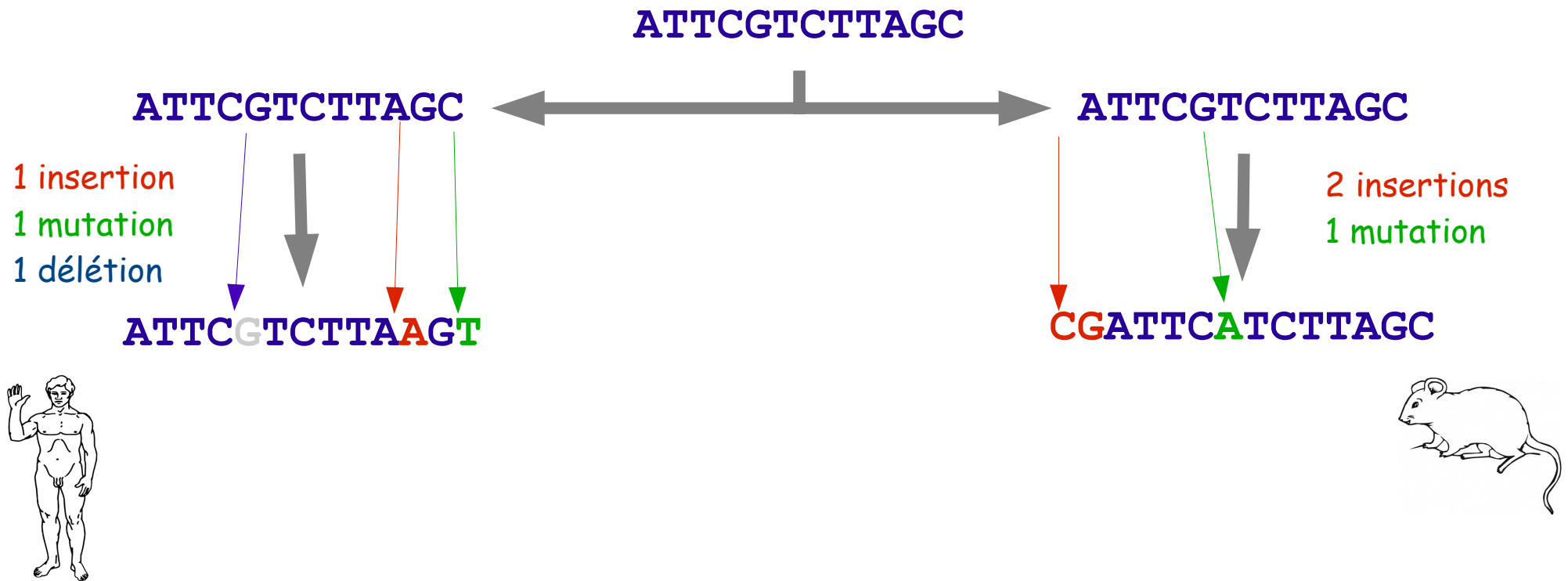


AACAGGGGCCCAATC

Alignement incorrect

CTGGGCCCAAGATC--  
AACAGGGGCCCAATC

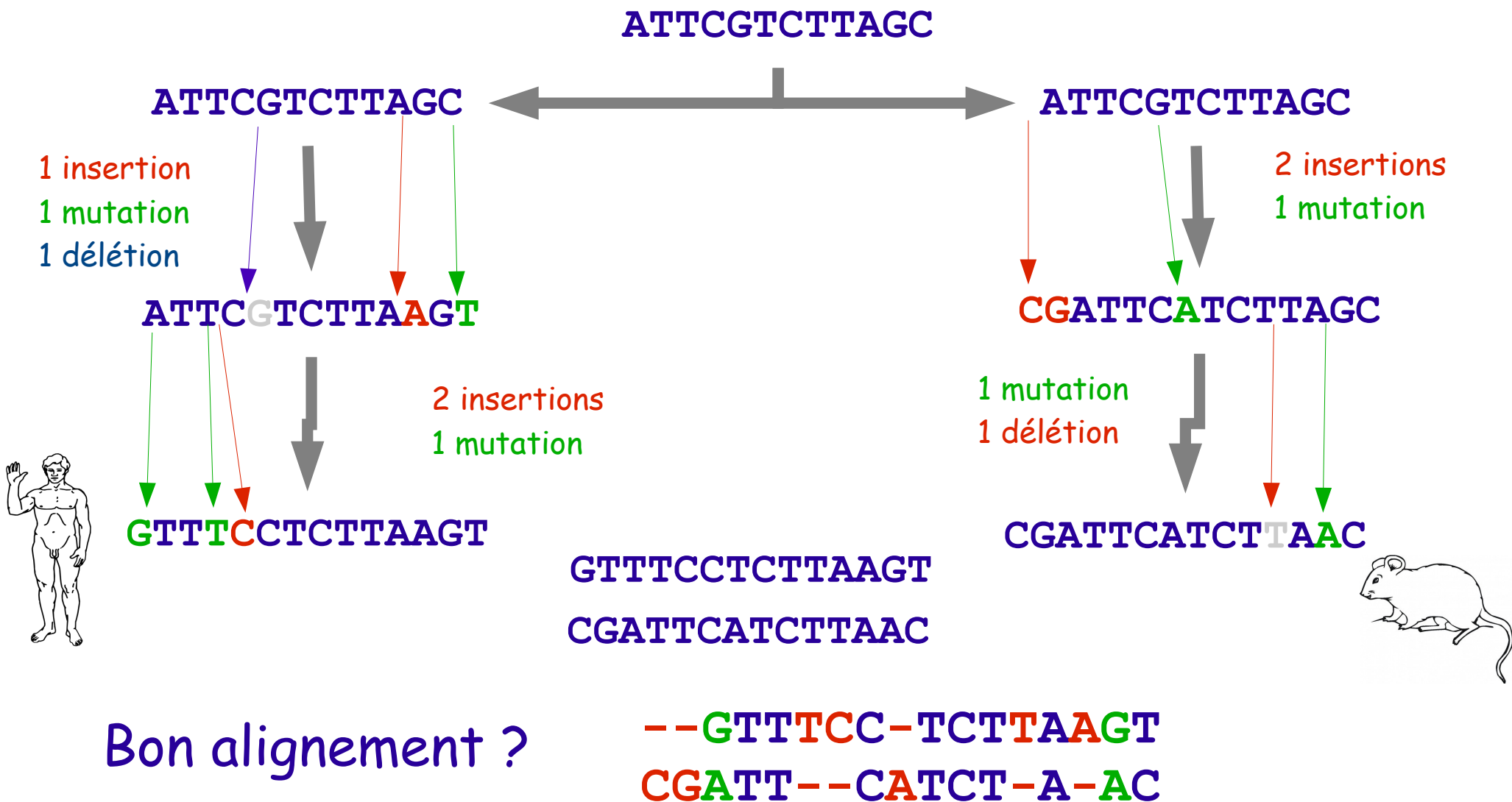
# Exercice



Bon alignement ?

--ATTCGTCTTAAGT  
CGATTCACTTA-GC

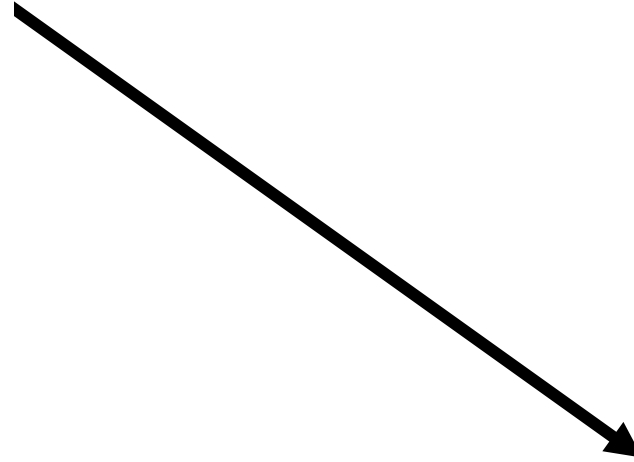
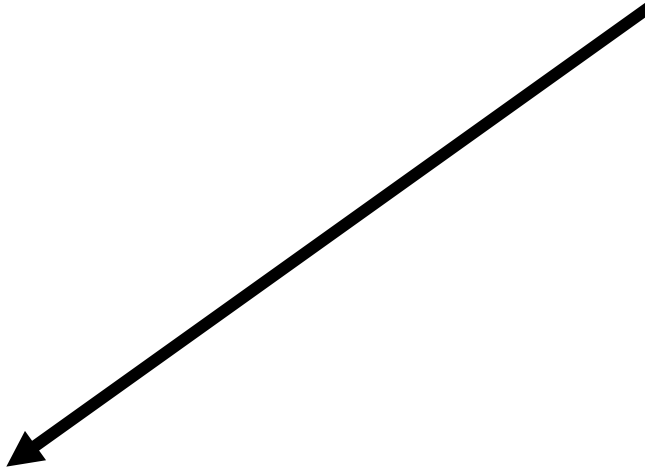
# Exercice



alignement des  
séquences



histoire évolutive des  
séquences



**CTGGGGCCCAGATC**

**AACAGGGGCCCAAATC**

**Alignement correct ?**

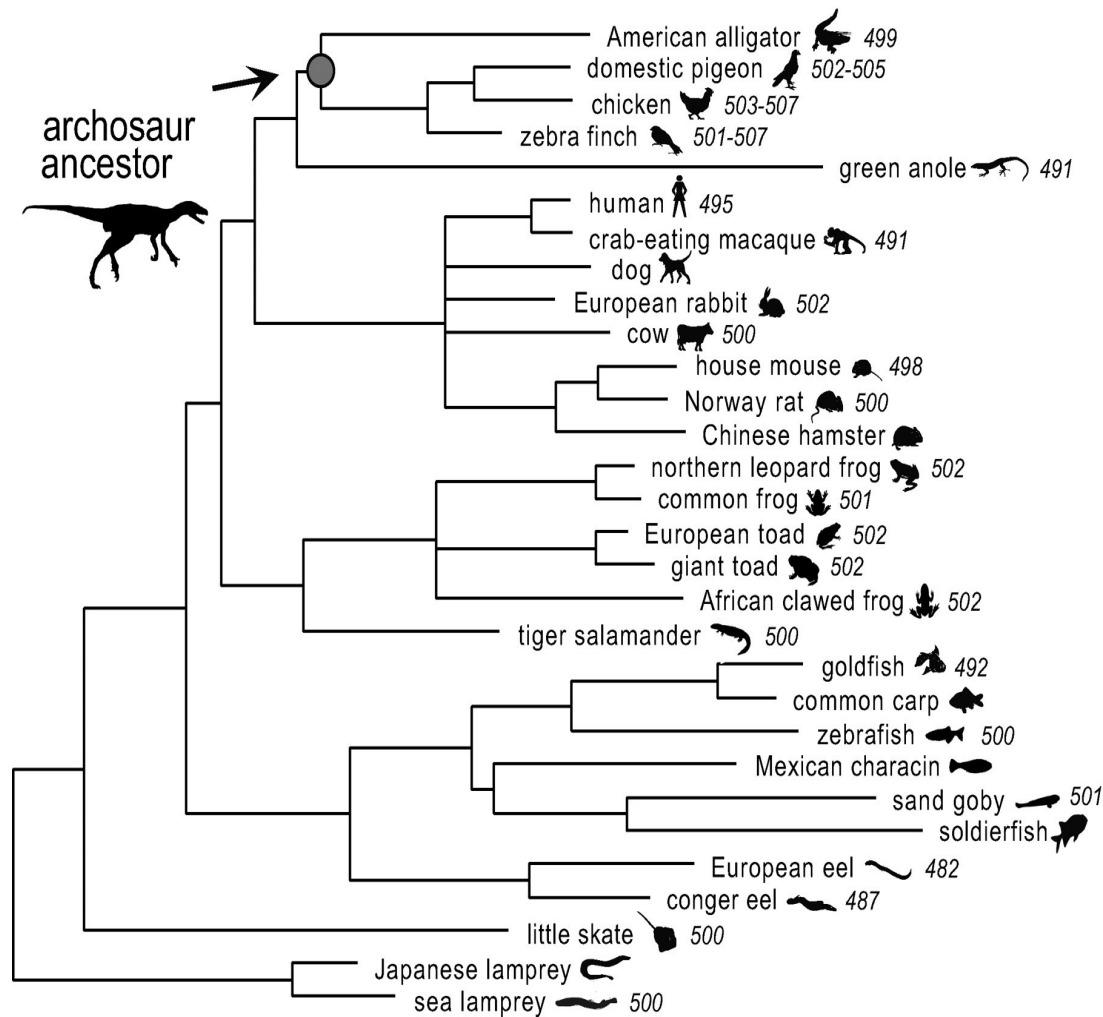
--CTGGGGCCCAGATC  
AACAGGGGCCCAAATC

**Alignement incorrect ?**

CTGGGGCCCAGATC--  
AACAGGGGCCCAAATC

# Comment voyaient les dinosaures ?

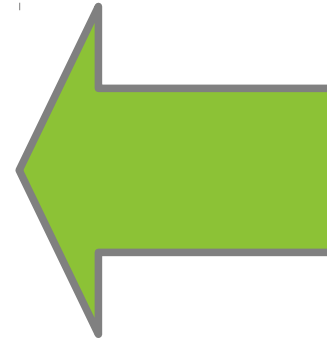
- peut-on retrouver quels étaient les pigments visuels de l'ancêtre des archosaures, à partir des séquences connues actuellement chez les vertébrés ?



Chang B S W et al. Mol Biol Evol 2002;19:1483-1489



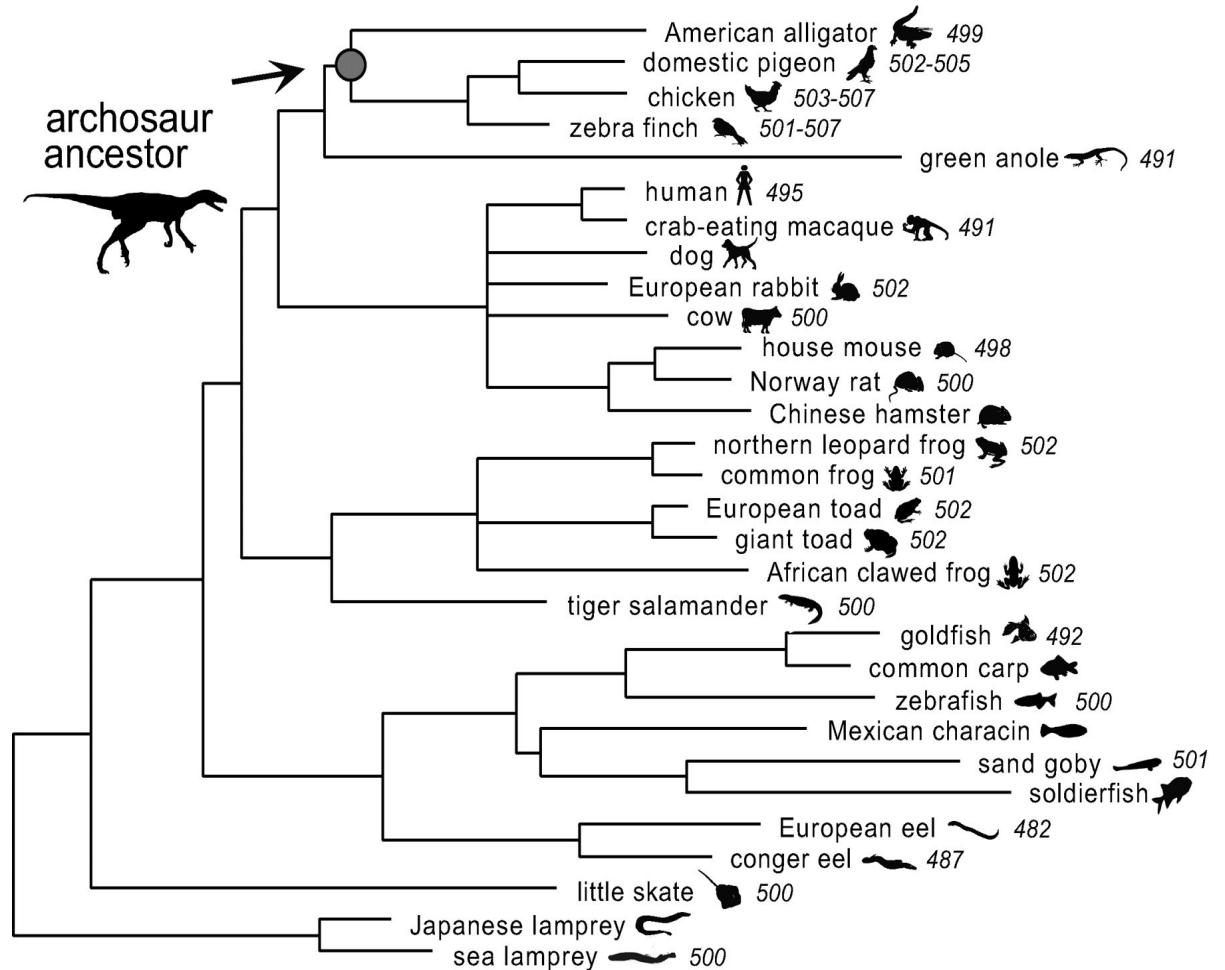
rhodopsines  
de dinosaure



rhodopsines  
vertébrés  
actuelles

>inferred ancient archosaur visual pigment

MNGTEGPNFYVPFSNKTGVVRSPFEYPQYYLAEPW  
QFSALAAYMFLLILLGFPINFLTLYVTIQHKKLRT  
PLNYILLNLAVADLFMVFGGFTTTMYTSMNGYFVF  
GPTGCNIEGFFATLGGEIALWSLVVLAIERVYVVC  
KPMSNFRFGENHAIMGVAFTWIMALACAAPPLFGW  
SRYIPEGMQCSCGVDYYTLKPEVNNESFVIYMFVV  
HFTIPLTVIFFCYGRLVCTVKEAAAQQQESATTQK  
AEKEVTRMVIIMVIAFLICWVPYASVAFYIFTHQG  
SDFGPIFMTIPAFFAKSSAIYNPVIYIVMNKQFRN  
CMITTLCCGKNPLGDDEASTTVSKTETSQVAPA

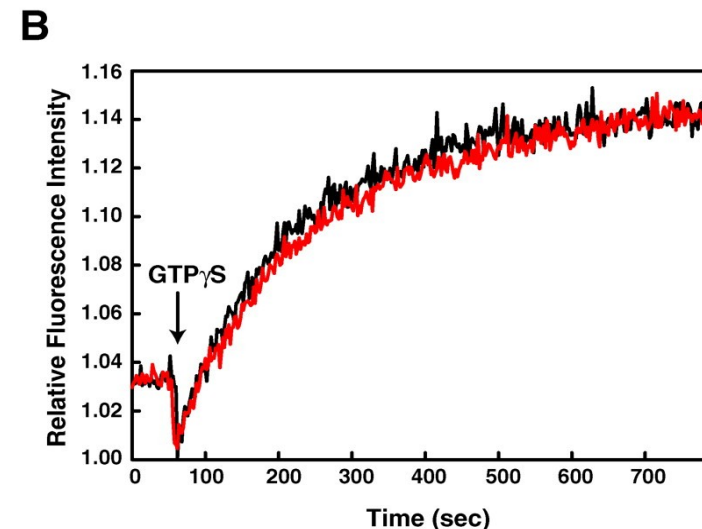
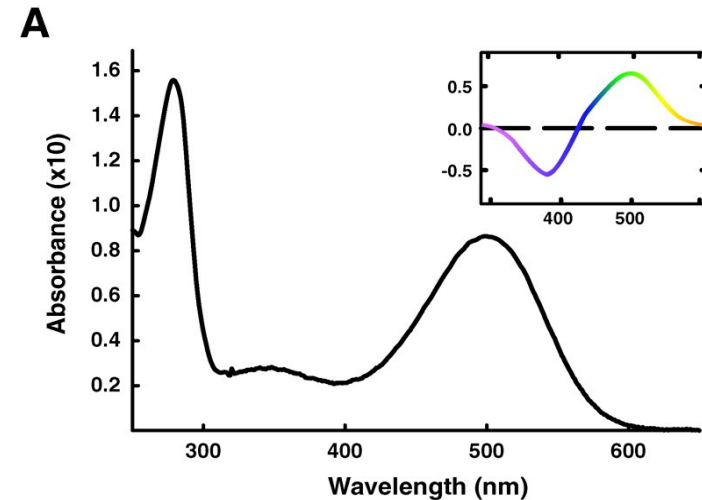


Chang B S W et al. Mol Biol Evol 2002;19:1483-1489



# Propriétés de la rhodopsine de dinosaure

- synthèse du gène correspondant à cette protéine
- transfection puis expression dans des lignées cellulaires
- purification de la protéine et étude de ses propriétés d'absorption



Chang B S W et al. Mol Biol Evol 2002;19:1483-1489





**Conclusion:**  
**les dinosaures voyaient rouge !**

## Plusieurs alignements possibles

CTGGGCCCCAGATC--  
 AACAGGGGCCCAAATC  
 . . . . \* . . \* . . \* . .

--CTGGGCCCCAGATC  
 AACAGGGGCCCAAATC  
 \* . \* \* \* \* \* \* \* . \* \* \*

CTGGGC--CCAGATC--  
 AACAGGGGCCCAA--TC  
 . . . . \* . \* \* \* . \*

### Quel est le bon ?

Principe de SIMPLICITE (=parcimonie)  
 le scénario le plus simple est le plus crédible

# Plusieurs alignements possibles

**MVPSYTR**  
**-MIPFPR**

mutations  $M \leftrightarrow V$ ,  $I \leftrightarrow P$ ,  
 $S \leftrightarrow P$ ,  $Y \leftrightarrow F$ ,  $T \leftrightarrow R$   
 et insertion/délétion de M

**MVPSYTR**  
**MIPF-PR**

mutations  $I \leftrightarrow V$ ,  $T \leftrightarrow P$  et  $S \leftrightarrow F$   
 et insertion/délétion de Y

**MVPSYTR**  
**MIP-FPR**

mutation  $I \leftrightarrow V$ ,  $T \leftrightarrow P$  et  $Y \leftrightarrow F$   
 et insertion/délétion de S

## Quel est le bon ?

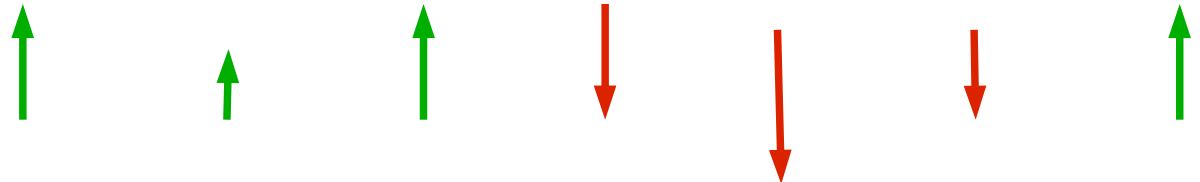
Principe de conservatisme  
 le scénario le plus fréquemment observé  
 est le plus crédible  
 ( $Y \leftrightarrow F$  est plus fréquent que  $S \leftrightarrow F$ )

# Principes du score

- principe de **PARCIMONIE**  
on pénalise tout évènement évolutif
  - insertion / délétion
  - mutations (la plupart ...)
- principe de **CONSERVATISME**
  - on pénalise les évènements **rarement observés**
  - on favorise le **conservatisme**

MVPSYTR  
 MIP-SCR

$$S = S(M, M) + S(V, I) + S(P, P) + S(-) + S(Y, S) + S(T, C) + S(R, R)$$





## alignement des séquences

**hypothèse:**  
alignement avec le  
score le plus élevé  
=  
scénario le plus probable



à défaut de connaître la  
vraie histoire, on va  
chercher l'alignement qui  
raconte l'histoire la plus **probable**

## histoire évolutive des séquences

# Alignements de séquences

## comment trouver le meilleur alignement?

**Idée numéro 1:**  
on les teste tous pour trouver le  
meilleur

pour 2 séquences de longueur:

( 1  $\mu$ s / alignement )

5 : 252 alignements

0.000252 s

10: 184756

0.18 s

20: 137846528820

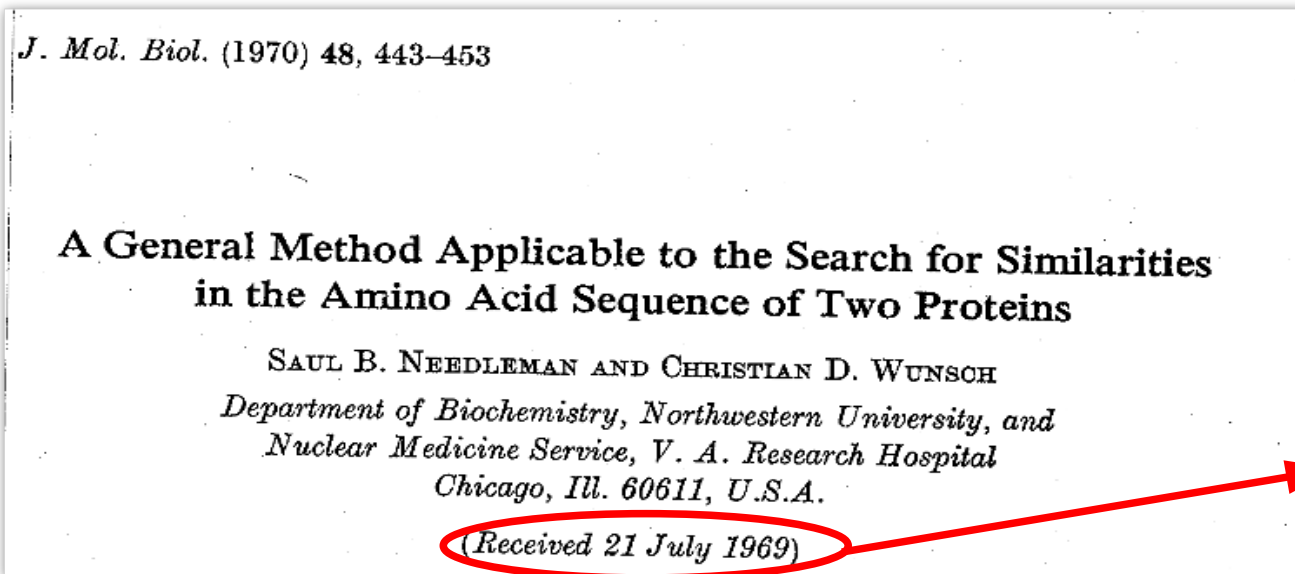
38 heures

50: 100891344545564193334812497256

31970537856352 siècles

**Idée numéro 2:**  
on trouve une meilleure idée ...

# Alignements de séquences *comment trouver le meilleur alignement?*

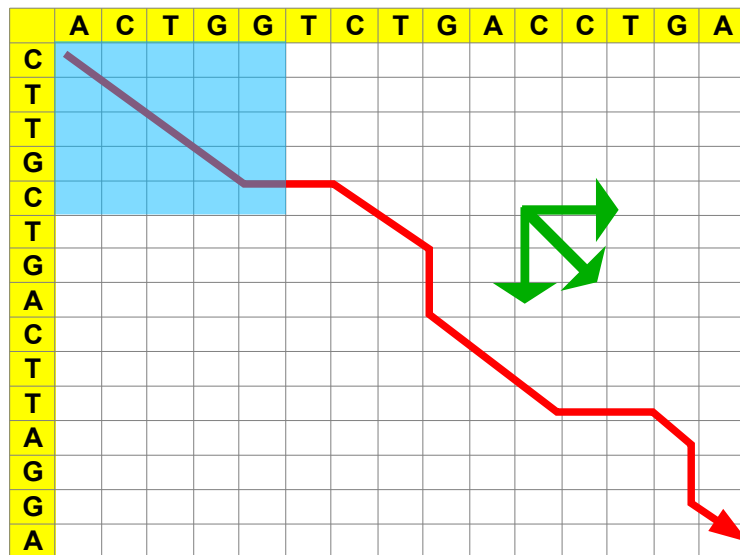


« *C'est un petit pas pour l'homme,  
mais un pas de géant pour la bioinformatique* »

# Alignements de séquences

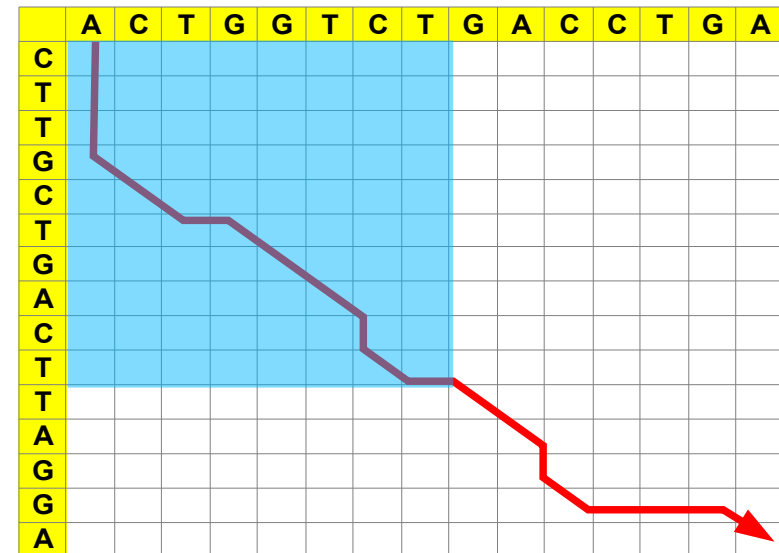
## Programmation dynamique

un alignement = un chemin dans la matrice



L=19

ACTGGTCT--GACCTG--A  
CTTG--CTGACTT--AGGA



---ACTGGT-CTGA-CCTGA  
CTTGC-TGACT-TAGG---A

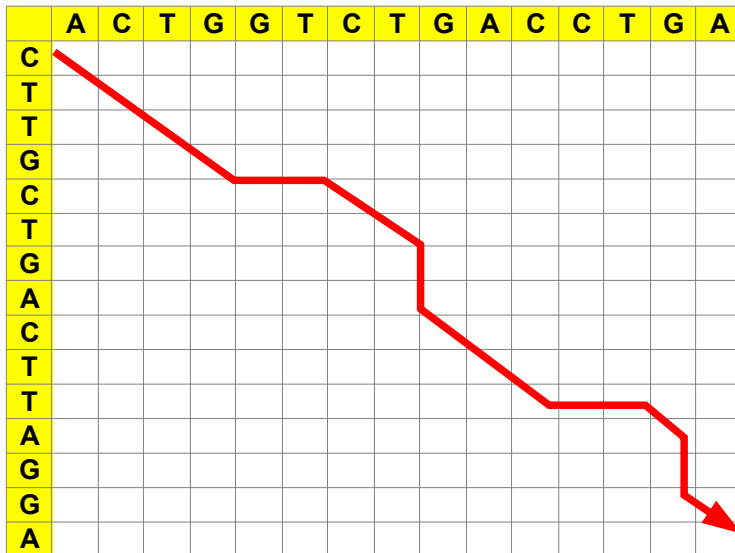
L=21

un pas supplémentaire = une position d'alignement en plus  
longueur du chemin = longueur de l'alignement

# Alignements de séquences

## Programmation dynamique

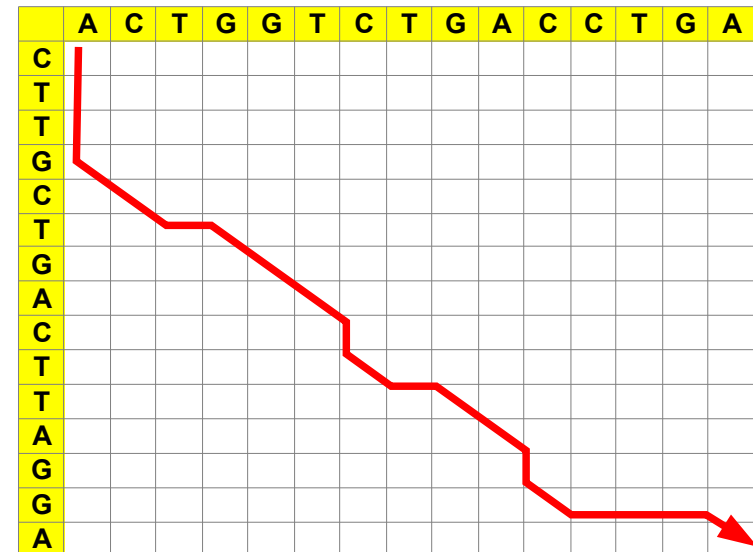
à chaque chemin (=alignement) est associé un score



match: 1  
 mismatch: 0  
 gap: -0.5

ACTGGTCT--GACCTG--A  
 CTTG--CTGACTT--AGGA

score = 1



---ACTGGT-CTGA-CCTGA  
 CTTGC-TGACT-TAGG---A

score = -1

On cherche le chemin correspondant au meilleur score.

# Alignements de séquences

## Programmation dynamique

Diviser...

...pour mieux aligner !

cet alignement de taille L aura le meilleur score ...

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | C | T | G | G | T | C | T | - | - | G | A | C | C | T | G | - | - | A |
| C | T | T | G | - | - | C | T | G | A | C | T | T | - | - | A | G | G | A |

... à condition que cet alignement de taille L-1  
ait le meilleur score !

# Alignements de séquences

## Programmation dynamique

Diviser...

...pour mieux aligner !

cet alignement de taille  $L-1$  aura le meilleur score ...

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | C | T | G | G | T | C | T | - | - | G | A | C | C | T | G | - | - | A |
| C | T | T | G | - | - | C | T | G | A | C | T | T | - | - | A | G | G | A |

... à condition que cet alignement de taille  $L-2$   
ait le meilleur score !

... etc ...

# Programmation dynamique

|   |  | A | C | T | G |
|---|--|---|---|---|---|
|   |  |   |   |   |   |
| C |  |   |   |   |   |
| F |  |   |   |   |   |
| F |  |   |   |   |   |
| G |  |   |   |   |   |



- Règle 1:**  
 chaque case va contenir un score; le score de l'alignement sera celui de la case en bas à droite
- Règle 2:**  
 le score d'une case se déduit à partir de celui des cases au-dessus, à gauche ou en diagonale
- Règle 3:**  
 un pas horizontal/vertical coûte 1 gap  
 un pas diagonal coûte 1 position alignée (match ou mismatch)



# Programmation dynamique

## Etape 1:

on remplit la première ligne et la première colonne

|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  |    |    |     |     |
| T | -8  |    |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

Score:  
gap: -4

# Programmation dynamique

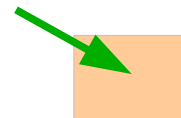
## Etape 2:

on remplit toutes les cases en gardant en mémoire le mouvement qui donne le meilleur score

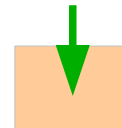
|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  |    |    |     |     |
| T | -8  |    |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

Score:

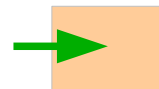
gap: -4    mismatch: -4



alignement AC → score =  $0 - 4 = -4$



insertion de gap → score =  $-4 - 4 = -8$



insertion de gap → score =  $-4 - 4 = -8$

# Programmation dynamique

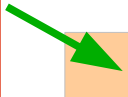
## Etape 2:

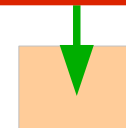
on remplit toutes les cases en gardant en mémoire le mouvement qui donne le meilleur score

|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  | -4 |    |     |     |
| T | -8  |    |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

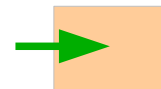
Score:

gap: -4    mismatch: -4


 alignement AC → score =  $0 - 4 = -4$



insertion de gap → score =  $-4 - 4 = -8$



insertion de gap → score =  $-4 - 4 = -8$

# Programmation dynamique

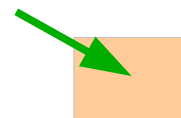
|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  | -4 | -8 |     |     |
| T | -8  |    |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

Score:

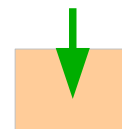
gap: -4

mismatch: -4

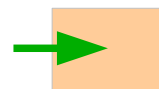
match: +4



alignement CC → score =  $-4 + 4 = 0$



insertion de gap → score =  $-8 - 4 = -12$



insertion de gap → score =  $-4 - 4 = -8$

# Programmation dynamique

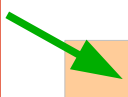
|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  | -4 | 0  |     |     |
| T | -8  |    |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

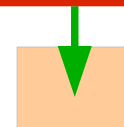
Score:


gap: -4

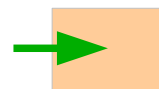
mismatch: -4


match: +4


 alignement CC → score =  $-4 + 4 = 0$




 insertion de gap → score =  $-8 - 4 = -12$




 insertion de gap → score =  $-4 - 4 = -8$

# Programmation dynamique

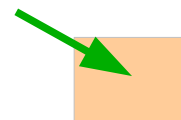
|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  | -4 | 0  |     |     |
| T | -8  |    |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

Score:

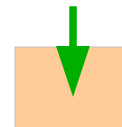
gap: -4

mismatch: -4

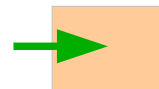
match: +4



alignement AT  $\rightarrow$  score =  $-4 - 4 = -8$



insertion de gap  $\rightarrow$  score =  $-4 - 4 = -8$



insertion de gap  $\rightarrow$  score =  $-8 - 4 = -12$

# Programmation dynamique


|   |     | A  | C  | T   | G   |
|---|-----|----|----|-----|-----|
|   | 0   | -4 | -8 | -12 | -16 |
| C | -4  | -4 | 0  |     |     |
| T | -8  | -8 |    |     |     |
| T | -12 |    |    |     |     |
| G | -16 |    |    |     |     |

Score:


gap: -4

mismatch: -4


match: +4



alignement AT → score =  $-4 - 4 = -8$



insertion de gap → score =  $-4 - 4 = -8$



insertion de gap → score =  $-8 - 4 = -12$

# Programmation dynamique

|   |     | A   | C   | T   | G   |
|---|-----|-----|-----|-----|-----|
|   | 0   | -4  | -8  | -12 | -16 |
| C | -4  | -4  | 0   | -4  | -8  |
| T | -8  | -8  | -4  | 4   | 0   |
| T | -12 | -12 | -8  | 0   | 0   |
| G | -16 | -16 | -12 | -4  | 4   |

Score:  
 gap: -4  
 mismatch: -4  
 match: +4

meilleur score



# Programmation dynamique

## Etape 3:

On part du score en bas à droite, et on remonte le cours des flèches pour trouver l'alignement (« backtracking »)

|   |     | A   | C   | T   | G   |
|---|-----|-----|-----|-----|-----|
|   | 0   | -4  | -8  | -12 | -16 |
| C | -4  | -4  | 0   | -4  | -8  |
| T | -8  | -8  | -4  | 4   | 0   |
| T | -12 | -12 | -8  | 0   | 0   |
| G | -16 | -16 | -12 | -4  | 4   |

2 chemins =  
2 alignements **optimaux**:

AC-TG  
-CTTG

ACT-G  
-CTTG

score: +4

### Bilan:

- 24 scores calculés
- $3^{4+4} = 6561$  chemins possibles

Alignement **global**:  
on aligne les 2 séquences  
du début à la fin

# Exercice

|   |  | C | C | T | G |
|---|--|---|---|---|---|
| C |  |   |   |   |   |
| T |  |   |   |   |   |
| T |  |   |   |   |   |
| C |  |   |   |   |   |

Aligner  
CCTG et CTTC

match : +4  
mismatch: -4

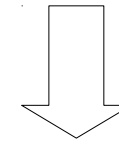
gap: -1 / -4 / -6

# Programmation dynamique

Est ce que cela marche pour les protéines ?

|   |  | M | G | K | P |
|---|--|---|---|---|---|
|   |  |   |   |   |   |
| R |  |   |   |   |   |
| N |  |   |   |   |   |
| I |  |   |   |   |   |
| L |  |   |   |   |   |
| V |  |   |   |   |   |

gap: -6  
match et mismatch



matrice de substitution  
(ex. BLOSUM62)

# Programmation dynamique

Est ce que cela marche pour les protéines ?

|   |     | M   | G   | K   | P   |
|---|-----|-----|-----|-----|-----|
|   | 0   | -6  | -12 | -18 | -24 |
| R | -6  | -1  | -7  | -10 | -16 |
| N | -12 | -7  | -1  | -7  | -12 |
| I | -18 | -11 | -7  | -4  | -10 |
| L | -24 | -16 | -13 | -9  | -7  |
| V | -30 | -22 | -19 | -15 | -11 |

gap: -6  
match et mismatch

MG - KP  
RNILV

# Programmation dynamique

|   |     | A  | T  | G  | C  | A  | T  | C  | C  | C  | A   | T   | G   | A   | C   |
|---|-----|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|
|   | 0   | -1 | -2 | -3 | -4 | -5 | -6 | -7 | -8 | -9 | -10 | -11 | -12 | -13 | -14 |
| T | -1  | -2 | 1  | 0  | -1 | -2 | -3 | -4 | -5 | -6 | -7  | -8  | -9  | -10 | -11 |
| C | -2  | -3 | 0  | -1 | 2  | 1  | 0  | -1 | -2 | -3 | -4  | -5  | -6  | -7  | -8  |
| T | -3  | -4 | -1 | -2 | 1  | 0  | 3  | 2  | 1  | 0  | -1  | -2  | -3  | -4  | -5  |
| A | -4  | -1 | -2 | -3 | 0  | 3  | 2  | 1  | 0  | -1 | 2   | 1   | 0   | -1  | -2  |
| T | -5  | -2 | 1  | 0  | -1 | 2  | 5  | 4  | 3  | 2  | 1   | 4   | 3   | 2   | 1   |
| A | -6  | -3 | 0  | -1 | -2 | 1  | 4  | 3  | 2  | 1  | 4   | 3   | 2   | 5   | 4   |
| T | -7  | -4 | -1 | -2 | -3 | 0  | 3  | 2  | 1  | 0  | 3   | 6   | 5   | 4   | 3   |
| C | -8  | -5 | -2 | -3 | 0  | -1 | 2  | 5  | 4  | 3  | 2   | 5   | 4   | 3   | 6   |
| C | -9  | -6 | -3 | -4 | -1 | -2 | 1  | 4  | 7  | 6  | 5   | 4   | 3   | 2   | 5   |
| G | -10 | -7 | -4 | -1 | -2 | -3 | 0  | 3  | 6  | 5  | 4   | 3   | 6   | 5   | 4   |
| T | -11 | -8 | -5 | -2 | -3 | -4 | -1 | 2  | 5  | 4  | 3   | 6   | 5   | 4   | 3   |

**Scores:**  
gap: -1  
mismatch: -3  
match: +2

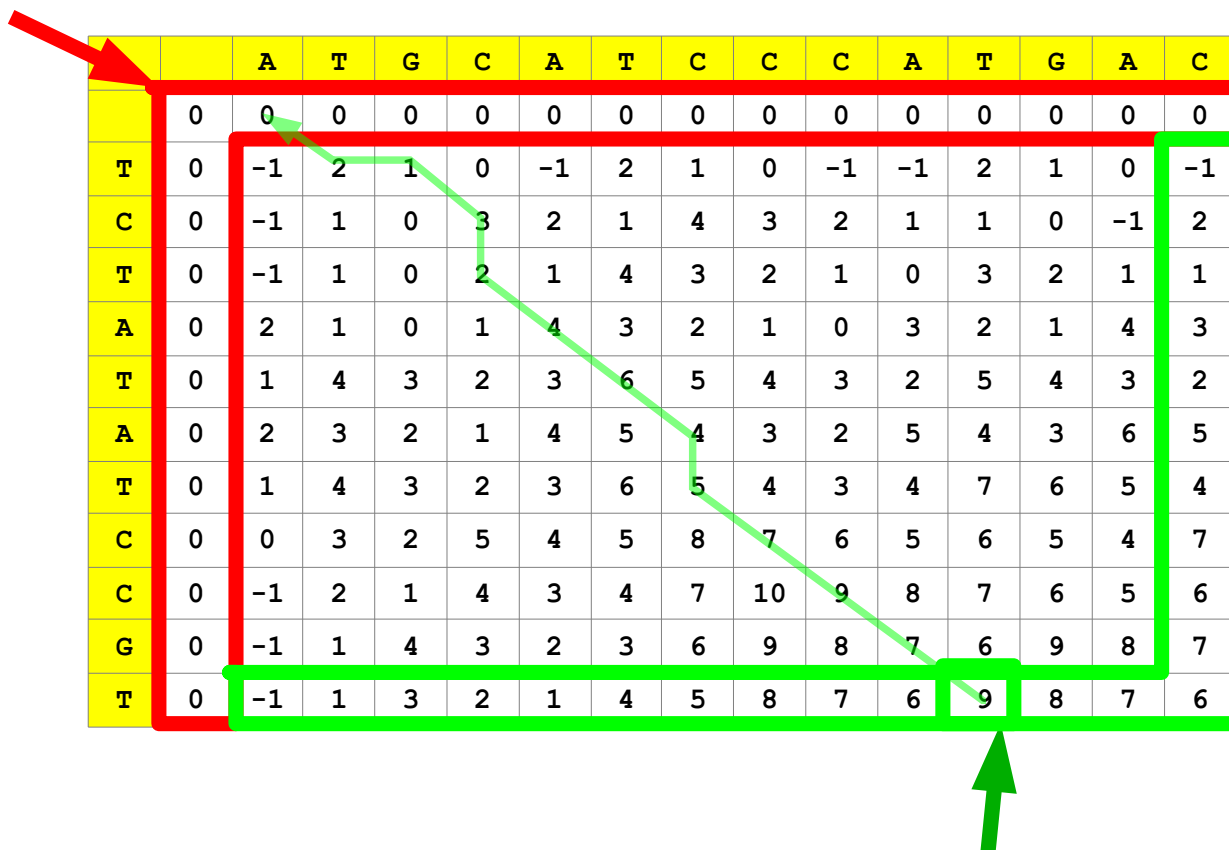
ATGC-ATC-CCATGAC  
-T-CTATATCCGT---

ces gaps coûtent cher, alors que les 2 séquences sont de longueurs différentes...

# Programmation dynamique

les gaps au début de chaque séquence ont un coût nul

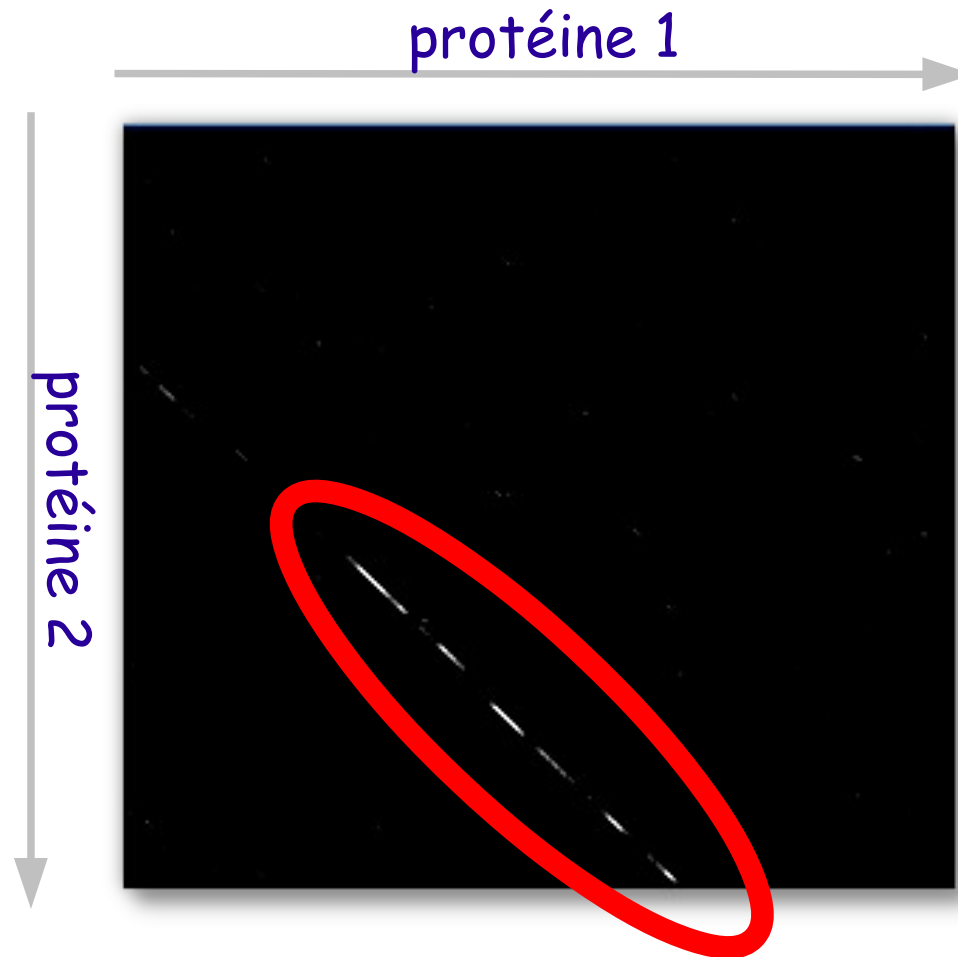
alignement semi-global:



|   |   | A  | T | G | C | A  | T | C | C  | C  | A  | T | G | A  | C  |
|---|---|----|---|---|---|----|---|---|----|----|----|---|---|----|----|
|   | 0 | 0  | 0 | 0 | 0 | 0  | 0 | 0 | 0  | 0  | 0  | 0 | 0 | 0  | 0  |
| T | 0 | -1 | 2 | 1 | 0 | -1 | 2 | 1 | 0  | -1 | -1 | 2 | 1 | 0  | -1 |
| C | 0 | -1 | 1 | 0 | 3 | 2  | 1 | 4 | 3  | 2  | 1  | 1 | 0 | -1 | 2  |
| T | 0 | -1 | 1 | 0 | 2 | 1  | 4 | 3 | 2  | 1  | 0  | 3 | 2 | 1  | 1  |
| A | 0 | 2  | 1 | 0 | 1 | 4  | 3 | 2 | 1  | 0  | 3  | 2 | 1 | 4  | 3  |
| T | 0 | 1  | 4 | 3 | 2 | 3  | 6 | 5 | 4  | 3  | 2  | 5 | 4 | 3  | 2  |
| A | 0 | 2  | 3 | 2 | 1 | 4  | 5 | 4 | 3  | 2  | 5  | 4 | 3 | 6  | 5  |
| T | 0 | 1  | 4 | 3 | 2 | 3  | 6 | 5 | 4  | 3  | 4  | 7 | 6 | 5  | 4  |
| C | 0 | 0  | 3 | 2 | 5 | 4  | 5 | 8 | 7  | 6  | 5  | 6 | 5 | 4  | 7  |
| C | 0 | -1 | 2 | 1 | 4 | 3  | 4 | 7 | 10 | 9  | 8  | 7 | 6 | 5  | 6  |
| G | 0 | -1 | 1 | 4 | 3 | 2  | 3 | 6 | 9  | 8  | 7  | 6 | 9 | 8  | 7  |
| T | 0 | -1 | 1 | 3 | 2 | 1  | 4 | 5 | 8  | 7  | 6  | 9 | 8 | 7  | 6  |

le score de l'alignement correspond au meilleur score de la dernière colonne ou la dernière ligne

# Similarité locale



il existe une similarité locale  
entre ces 2 séquences

# Programmation dynamique

|   |   | A  | T | G | C | A  | T | C | C         | C  | A  | T        | G | A  | C  |
|---|---|----|---|---|---|----|---|---|-----------|----|----|----------|---|----|----|
|   | 0 | 0  | 0 | 0 | 0 | 0  | 0 | 0 | 0         | 0  | 0  | 0        | 0 | 0  | 0  |
| T | 0 | -1 | 2 | 1 | 0 | -1 | 2 | 1 | 0         | -1 | -1 | 2        | 1 | 0  | -1 |
| C | 0 | -1 | 1 | 0 | 3 | 2  | 1 | 4 | 3         | 2  | 1  | 1        | 0 | -1 | 2  |
| T | 0 | -1 | 1 | 0 | 2 | 1  | 4 | 3 | 2         | 1  | 0  | 3        | 2 | 1  | 1  |
| A | 0 | 2  | 1 | 0 | 1 | 4  | 3 | 2 | 1         | 0  | 3  | 2        | 1 | 4  | 3  |
| T | 0 | 1  | 4 | 3 | 2 | 3  | 6 | 5 | 4         | 3  | 2  | 5        | 4 | 3  | 2  |
| A | 0 | 2  | 3 | 2 | 1 | 4  | 5 | 4 | 3         | 2  | 5  | 4        | 3 | 6  | 5  |
| T |   |    |   |   |   | 3  | 6 | 5 | 4         | 3  | 4  | 7        | 6 | 5  | 4  |
| C |   |    |   |   |   | 4  | 5 | 8 | 7         | 6  | 5  | 6        | 5 | 4  | 7  |
| C |   |    |   |   |   | 3  | 4 | 7 | <b>10</b> | 9  | 8  | 7        | 6 | 5  | 6  |
| G |   |    |   |   |   | 2  | 3 | 6 | 9         | 8  | 7  | 6        | 9 | 8  | 7  |
| T |   |    |   |   |   | 1  | 4 | 5 | 8         | 7  | 6  | <b>9</b> | 8 | 7  | 6  |

il y a un score meilleur ici:

alignement local

ATCC  
ATCC

alignement  
(semi) global

ATGC-ATC-CCATGAC  
-T-CTATATCCGT---



# Programmation dynamique

## alignement local (Smith-Waterman)

|   |   | A  | T | G | C | A  | T | C | C  | C  | A  | T | G | A  | C  |
|---|---|----|---|---|---|----|---|---|----|----|----|---|---|----|----|
|   | 0 | 0  | 0 | 0 | 0 | 0  | 0 | 0 | 0  | 0  | 0  | 0 | 0 | 0  | 0  |
| T | 0 | -1 | 2 | 1 | 0 | -1 | 2 | 1 | 0  | -1 | -1 | 2 | 1 | 0  | -1 |
| C | 0 | -1 | 1 | 0 | 3 | 2  | 1 | 4 | 3  | 2  | 1  | 1 | 0 | -1 | 2  |
| T | 0 | -1 | 1 | 0 | 2 | 1  | 4 | 3 | 2  | 1  | 0  | 3 | 2 | 1  | 1  |
| A | 0 | 2  | 1 | 0 | 1 | 4  | 3 | 2 | 1  | 0  | 3  | 2 | 1 | 4  | 3  |
| T | 0 | 1  | 4 | 3 | 2 | 3  | 6 | 5 | 4  | 3  | 2  | 5 | 4 | 3  | 2  |
| A | 0 | 2  | 3 | 2 | 1 | 4  | 5 | 4 | 3  | 2  | 5  | 4 | 3 | 6  | 5  |
| T | 0 | 1  | 4 | 3 | 2 | 3  | 6 | 5 | 4  | 3  | 4  | 7 | 6 | 5  | 4  |
| C | 0 | 0  | 3 | 2 | 5 | 4  | 5 | 8 | 7  | 6  | 5  | 6 | 5 | 4  | 7  |
| C | 0 | -1 | 2 | 1 | 4 | 3  | 4 | 7 | 10 | 9  | 8  | 7 | 6 | 5  | 6  |
| G | 0 | -1 | 1 | 4 | 3 | 2  | 3 | 6 | 9  | 8  | 7  | 6 | 9 | 8  | 7  |
| T | 0 | -1 | 1 | 3 | 2 | 1  | 4 | 5 | 8  | 7  | 6  | 9 | 8 | 7  | 6  |

- à chaque étape du remplissage de la matrice, si le score devient négatif, on le met à 0
- on revient sur ses pas à partir du score le plus élevé, et on s'arrête avant de passer à 0

# Programmation dynamique

- Needleman & Wunsch
  - **global**: le meilleur score est celui de la case en bas à droite
  - **semi-global**: le meilleur score est le score le plus élevé sur la dernière ligne ou colonne
- Smith & Waterman
  - **local**: on part du score le plus élevé de la matrice

**pour répondre à la question**

"quelle est la **similarité** entre 2 séquences ?  
et donc: est-ce que ces deux séquences sont  
**homologues** ?"

# Efficace ?

| Longueur des séquences | Enumération                | Programmation dynamique (Smith-waterman) |
|------------------------|----------------------------|--|
| 5                      | 0.000252 s                 |  |
| 10                     | 0.18 s                     |  |
| 20                     | 38 heures                  |  |
| 50                     | 3197053785635<br>2 siècles | 0.018 s                                  |
| 350                    | hahahahaha !!!             | 0.035 s                                  |

*C'est clairement mieux, mais est ce que c'est suffisant ??*

"quelle est la **similarité** entre ces 2 séquences ?  
et donc: est-ce que ces deux séquences sont  
**homologues** ?"

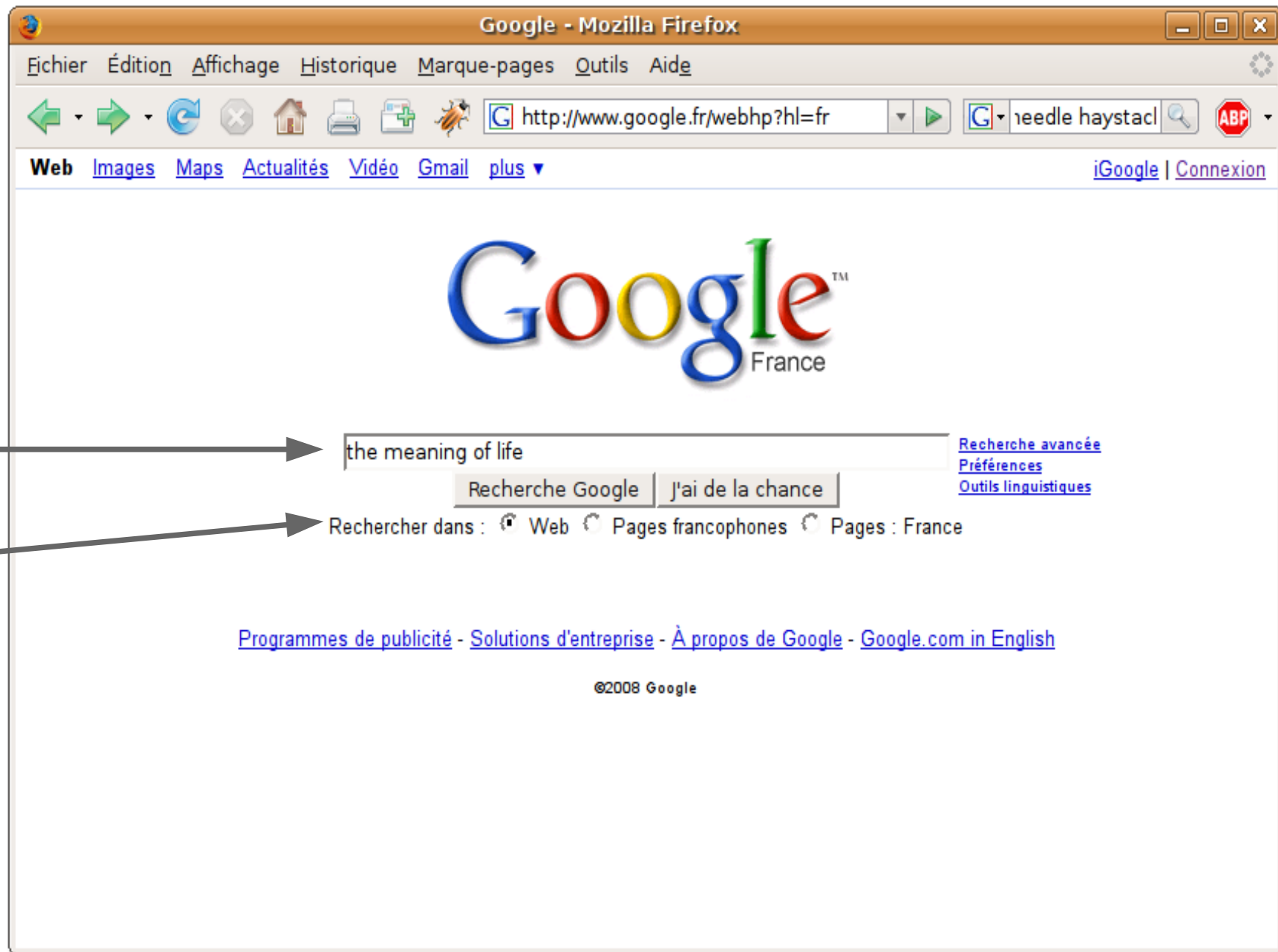


"existe-t-il des séquences homologues à la mienne  
**parmi toutes les séquences connues** ? "  
(ex: UniProt, ~ 12 millions de séquences, ~ 350 AA/seq)

**Smith & Waterman:**

**$0.035 \text{ s} \times 12 \text{ millions} = \dots 118 \text{ heures} \sim 5 \text{ jours} !!$**

# Google = fouiller l'Internet



Requête

Portée

# BLAST = fouiller les séquences biologiques

Requête

Portée

